DETAILED ACTION

This office action is responsive to communication filed March 2, 2010. Claims 1, 4, 5, 8 and 9 are currently amended. Claims 1-20 remain pending in this application.

Information Disclosure Statement

1. The information disclosure statement (IDS) submitted on March 2, 2010 is in compliance with the provisions of 37 CFR 1.97. Accordingly, the information disclosure statement is being considered by the examiner.

Claim Objections

2. Claim 4 is objected to because of the following informalities: The Examiner believes that the word "identify" has been mistakenly deleted from line 6 of the claim because the claim does not make sense without the word. Appropriate correction is required.

3. Claim 8 is objected to because of the following informalities: The Examiner believes that the word "identifying" has been mistakenly deleted from line 6 of the claim because the claim does not make sense without the word. Appropriate correction is required.

Claim Rejections - 35 USC § 112

4. The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

5. Claims 4 and 8-12 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

6. Claims 4 and 8 each recite "a one-to-one relationship between the first child object and the second child object, wherein the first child object and the second child object are at the same hierarchical level within the hierarchical structure." However, this limitation is ambiguous. It is unclear as to how two child objects at the same level can have a one-to-one relationship.

7. In accordance with the specification and what is known to one having ordinary skill in the art, a one-to-one relationship normally refers to objects at different levels of a hierarchy, and not the objects on the same level. Generally, objects on the same level are not related to each other, instead an object is generally related to a parent object and/or child object on the level above and/or beneath the object.

8. Furthermore, the specification teaches collapsing and flattening a hierarchy by merging objects and promoting objects into the parent object (Figs. 16 and 17, page 20 line 27 - page 21 line12). This supports the idea of one-to-one relationships between objects on different levels and not objects on the same level. In fact, there is no support in the specification for child objects on the same level having a one-to-one relationship. Further, the limitation "moving a content of the first child object into the second child object at the same hierarchical level within the hierarchical structure" is unclear for similar reasons. Therefore, claims 4 and 8 are indefinite.

Claim Rejections - 35 USC § 102

9. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

10. Claims 1-3, 5-7, and 13-20 are rejected under 35 U.S.C. 102(e) as being anticipated by Cutlip (US 2004/0039738 A1).

With respect to claims 1 and 5, Cutlip teaches a computer-implemented method for use in a web services system having complex UDDI object(s) (paragraphs 4 and 24), the method comprising:

providing a database for storing at least one directory parent object within a first object class, the at least one directory parent object storing a plurality of attributes, wherein the at least one direct parent object stores at least one unique attribute (i.e. Dun and Bradstreet Number, US Tax Identifier, and NAICS classification) that occurs only once in the at least one directory parent object (Fig. 10, paragraphs 45-46) and a repeating attribute that occurs more than once in the at least one directory parent object (Figs. 3, 15A and 15B, paragraphs 40, 43, 45 and 70);

using a processor in communication with the database to create a first directory child object for storing a first value associated with the repeating attributes, the first

directory child object also within the first object class (Figs. 3, 15A and 15B, paragraphs 40, 43 and 70);

using the processor to remove the repeating attribute from the at least one directory parent object such that the at least one directory parent object comprises only unique attributes (Figs. 3 and 10, paragraphs 40, 34 and 45-46); and

storing, in the database, the value associated with the repeating attribute in the first directory child object (Figs. 3, 15A and 15B, paragraphs 40, 43 and 70).

With respect to claims 2 and 6, Cutlip teaches using the processor to create a second directory child object (*i.e. Acme Service Number 2*) for storing a second value associated with the repeating attribute, the second child object also within the first object class (Figs. 3, 15A and 15B, paragraphs 40, 43 and 70).

With respect to claims 3 and 7, Cutlip teaches wherein the parent object is at least one of a business entity, business service, binding template and tmodel (Fig. 3, paragraph 43).

With respect to claims 13 and 17, Cutlip teaches further comprising creating a searchable index of the first value associated with the repeating attribute (paragraphs 22 and 76, claim 3).

With respect to claims 14 and 18, Cutlip teaches storing at least one unique attribute in the directory parent object (paragraph 43).

With respect to claims 15 and 19, Cutlip teaches wherein the directory parent object comprises a business entity and the at least one unique attribute comprises a business key (Figs. 3 and 10, paragraphs 43 and 45).

With respect to claims 16 and 20, Cutlip teaches wherein the first directory child object is selected from the group consisting of name, description, contact, discovery URL, keyed references and business services (Fig.3, paragraphs 42-43).

11. Claims 1-3 and 5-7 are rejected under 35 U.S.C. 102(e) as being anticipated by Gadbois et al. (US 2004/0002955 A1) ('Gadbois').

With respect to claims 1 and 5, Gadbois teaches a method for use in a web services system having complex UDDI object(s) (paragraphs 21 and 24), the method comprising:

providing a database for storing at least one directory parent object within a first object class, the at least one directory parent object storing a plurality of attributes, wherein the at least one direct parent object stores at least one unique attribute (*i.e. organization name*) that occurs only once in the at least one directory parent object and

a repeating attribute that occurs more than once in the at least one directory parent object (Fig. 2, paragraphs 27-28);

using a processor in communication with the database to create a first directory child object for storing a first value associated with the repeating attributes, the first directory child object also within the first object class (Fig. 2, paragraph 28);

using the processor to remove the repeating attribute from the at least one directory parent object such that the at least one directory parent object comprises only unique attributes (Fig. 2, paragraphs 27-28); and

storing, in the database, the value associated with the repeating attribute in the first directory child object (Fig. 2, paragraphs 28).

With respect to claims 2 and 6, Gadbois teaches using the processor to create a second directory child object (*i.e. Business Service 2*) for storing a second value associated with the repeating attribute, the second child object also within the first object class (Fig. 2, paragraph 28).

With respect to claims 3 and 7, Gadbois teaches wherein the parent object is at least one of a business entity, business service, binding template and tmodel (Fig. 2, paragraph 27).

Claim Rejections - 35 USC § 103

12. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

13. Claims 4 and 8-12 are rejected under 35 U.S.C. 103(a) as being unpatentable over Gadbois et al. (US 2004/0002955 A1) ('Gadbois') in view of Colgan (US 5,956,499), and further in view of Sutherland (US 7,054,858 B2)

With respect to claims 4 and 8, Gadbois teaches a method of flattening a hierarchy in a web services arrangement, the method comprising:

providing a database for storing a hierarchical structure of a plurality of UDDI objects, the plurality of objects comprising at least one parent object, at least a first child object, and at least a second child object (Fig. 2, paragraphs 21-23); and

using a processor in communication with the database to identify at most a one-to-one relationship between the first child object and the second child object, wherein the first child object and the second child object are at the same hierarchical level (elements 262 and 272 in Figure 2).

Gadbois does not teach using the processor to remove a portion of the hierarchical structure having the one-to-one relationship by moving content of the first child object into the second child object at the same hierarchical level within the hierarchical structure.

Colgan teaches a method and system for non-model based application transitioning (see abstract), in which he teaches determining a one-to-one relationship between objects (col. 2 lines 47-62 and Table 1; Fig. 4A); and using the processor to remove a portion of the hierarchical structure having the one-to-one relationship by moving content of the first child object (*customer address*) into a second object (*customer*) at the same hierarchical level within the hierarchical structure (column 3 lines 1-12 and Table 2; Fig. 4B).

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to have modified Gadbois by the teaching of Colgan because using the processor to remove a portion of the hierarchical structure having the one-to-one relationship by moving content of the first child object into the second child object would enable a reduction in the number of entities in a model or hierarchy and thus make the model or hierarchy easier to work with (Colgan, column 2 lines 47-60).

Further regarding claims 4 and 8, it is inherent that any object may act as a parent object or child object. However, Gadbois in view of Colgan fails to explicitly teach that the second object (i.e. customer) is a child object.

Sutherland teaches object to relational database mapping (see abstract), in which he teaches a hierarchy having a first parent object (*i.e. order*), a first child object (*i.e. customer address*) and a second child object (*i.e. customer*) (Fig. 1A).

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to have further modified Gadbois by the teaching of Sutherland

because the second object (i.e. customer) being a child object would enable efficient retrieval of objects having nested relationships (Sutherland, abstract).

With respect to claim 9, Gadbois as modified teaches wherein the at least one parent object is at least one of a business entity, business service, binding template and tmodel (Gadbois, Figure 2, paragraphs 27-28).

With respect to claims 10 and 12, Gadbois as modified teaches wherein the first child object is a relationship object (Gadbois, Figure 2; Colgan, Fig. 4A-B).

With respect to claim 11, Gadbois as modified teaches wherein the parent object is at least one of a business entity, business service, binding template and tmodel (Gadbois, Figure 2, paragraphs 27-28; Colgan, Fig. 4A; Sutherland, Fig. 1A).

Response to Arguments

14. Applicant's arguments filed March 2, 2010 have been fully considered but they are not persuasive.

15. Applicant argues that Cutlip and Gadbois each fail to teach storing a repeating attribute. Examiner disagrees. Both Cutlip and Gadbois teach directory parent objects having repeating attributes and creating a directory child object associated with the repeating attributes. For example, Cutlip teaches a directory parent object business entity 320, which has 1 to many discovery URLS, one to many contacts and zero to

many business services (Figure 3). Furthermore, Figures 15A and 15B represent a business entity having two different business services. Thus it is clear that the parent object has repeating attributes (evidenced by the one to many relationships), and a value associated with the repeating attributes are stored as directory child objects.

16. Applicant argues that the relationship between the objects does not speak to the attributes that are stored in the Business Entity object. However this is incorrect. The child objects having one to many relationships may represent repeating attributes. The repeating attributes are represented as child objects because of the fact that they repeat or have a one to many relationship. As explained below, instead of having repeating attributes, the attributes are represented as child objects.

17. Likewise Gadbois teaches directory parent objects (i.e. business services node 242) having repeating attributes (i.e. business service1 and business service 2) and creating a directory child object associated with the repeating attributes (i.e. child nodes 243 and 244). Thus it is clear that the parent object of Gadbois also has repeating attributes (evidenced by the one to many relationships), and a value associated with the repeating attributes are stored as directory child objects.

18. Applicant also argues that the prior fails to teach at least one unique attribute that occurs only once in the at least one directory parent object. Examiner disagrees. Cutlip teaches that a business entity (i.e. a directory parent object) may be uniquely identified by either a Dun and Bradstreet Number, US Tax Identifier, or by NAICS classification (Fig. 10, paragraphs 45-46). Thus any of these three attributes represent unique attributes of the directory parent object. Similarly, Gadbois teaches that an organization

name (i.e. Organization1) may be used to uniquely identify an organization parent object. Thus, Gadbois also teaches the directory parent object having at least one unique attribute that occurs only once in the directory parent object.

19. Applicant further argues that the prior art fails to teach removing the repeating attribute from the directory parent object such that the directory parent object comprises only unique attributes. Examiner disagrees. Both Cutlip and Gadbois teach that repeating attributes are moved into child objects and thus the directory parent object is left with only unique attributes. For example, Cutlip shows (Fig. 10) that a business entity (i.e. parent object) has three attributes (Dun and Bradstreet Number, US Tax Identifier, and NAICS classification), which are all unique attributes. The other attributes are all represented as child objects having one-to-many or zero-to-many relationships with the business entity (Fig. 3). The same follows with the directory of Gadbois, as shown in Fig. 2.

20. Although the references do not explicitly recite the language "removing the repeating attributes," it is clear that repeating attributes are represented as child objects, and are thus removed from the directory parent object. It is well known in the art that it is more efficient to store repeating attributes as child objects, leaving parent objects with only unique attributes.

21. Applicant further argues that Gadbois in view of Colgan fails to teach using a processor in communication with the database to identify at most a one-to-one relationship between the first child object and the second child object, wherein the first child object and the second child object are at the same hierarchical level. The

Examiner disagrees. As explained in the 35 U.S.C 112 rejection, this limitation is indefinite. However, Gadbois teaches one-to-one relationships between objects (Figure 2). Because the claim limitation is unclear, it can be argued that the relationship between groups 262 and business services 272 is a one-to-one relationship between a first child object and a second child object at the same hierarchical level.

22. Applicant further argues that there is no disclosure of using a processor to identify the relationship between objects. Examiner disagrees. Gadbois teaches that a directory server, which includes an information processing system such as processing hardware (paragraph 23), stores information in a tree-like structure such as a directory information tree (DIT). He further teaches that the DIT is populated by a plurality of nodes and object classes are used to define the nodes (paragraph 26). Thus it is clear that a processor is used in identifying and storing objects and relationships between them. However, automating a manual activity or process alone is not sufficient to distinguish over the prior art (see MPEP 2144.04).

23. Lastly, Applicant argues that Colgan does not teach moving content of a child object into a different child object. Colgan has been used to teach removing a portion of a hierarchical structure by moving content of a first child object into a second object. The Examiner admits that Colgan does not teach that the second or customer object is a child object and uses Sutherland to show that a customer object may be represented as a child object. In response to applicant's arguments against the references individually, one cannot show nonobviousness by attacking references individually where the rejections are based on combinations of references. See *In re Keller*, 642

F.2d 413, 208 USPQ 871 (CCPA 1981); *In re Merck & Co.*, 800 F.2d 1091, 231 USPQ 375 (Fed. Cir. 1986).

Conclusion

24. Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to ALICIA M. LEWIS whose telephone number is (571)272-5599. The examiner can normally be reached on Monday - Friday, 9 - 6:30, alternate Friday off.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Charles Rones can be reached on 571-272-4085. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/A. M. L./
Examiner, Art Unit 2164
June 15, 2010

/Charles Rones/
Supervisory Patent Examiner, Art Unit 2164